

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号  
特開2000-152244  
(P2000-152244A)

(43)公開日 平成12年5月30日(2000.5.30)

(51)Int.Cl.<sup>7</sup>  
H 0 4 N 7/32

識別記号

F I  
H 0 4 N 7/137

テーマコード\* (参考)  
Z 5 C 0 5 9

審査請求 未請求 請求項の数6 O L (全 9 頁)

(21)出願番号 特願平10-320760

(22)出願日 平成10年11月11日(1998.11.11)

(71)出願人 000003821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(72)発明者 井戸 大治

神奈川県横浜市港北区綱島東四丁目3番1

号 松下通信工業株式会社内

(74)代理人 100105050

弁理士 鷲田 公一

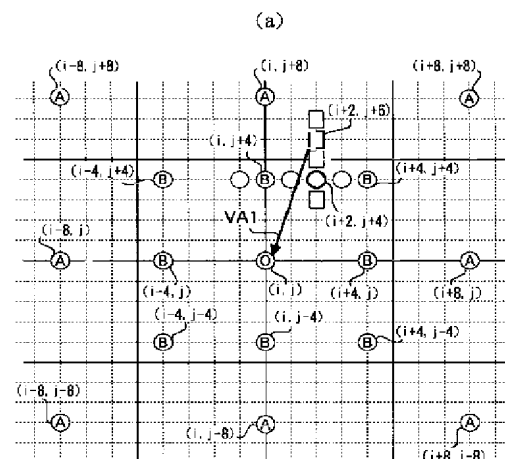
Fターム(参考) 5C059 KK19 MA05 MA23 NN05 NN10  
NN11 NN28 NN32

(54)【発明の名称】 動きベクトル検出方法および画像符号化方法

(57)【要約】

【課題】 動きベクトルが大きい場合でも、簡易な方法を用いてその動きベクトルを的確かつ効率的に検出すること。

【解決手段】 one-at-a-time探索を行う場合、着目するブロック(i, j)と同じ位置にある前フレームのブロックとの相似度を算出し、その相似度を参照して評価値としてクラス分けを行い、そのクラス分けに応じて、探索範囲と探索開始候補のブロックを適宜選択する。



(b)

| 差分値    | 探索開始候補 |
|--------|--------|
| 10以下   | 原点     |
| 100以下  | B群     |
| 1000以下 | A群     |

## 【特許請求の範囲】

【請求項1】 現フレームの注目するブロックと、このブロックと同位置にある前フレームのブロックとの相似度を算出し、算出された相似度の程度に応じて、動きベクトルの探索開始候補となるブロックを変更することを特徴とする動きベクトル検出方法。

【請求項2】 符号化対象フレームを複数のブロックに分割し、着目するブロックについて、前フレームの探索範囲内にあるブロックとの間の相似度を判定し、最も高い相似度を示すブロックの位置が、前記着目するブロックから  $(x, y)$  だけシフトしている場合に、そのシフト量  $v = (x, y)$  を動きベクトルとする、ブロックマッチング法を用いた動きベクトル検出方法において、前記着目するブロックと同じ位置にある前フレームのブロックについて相似度を求め、その相似度のレベルを所定の基準により判定し、相似度が低い場合には、相似度が高い場合に比べて、one-at-a-time探索の始点となる可能性をもつ複数の探索開始候補ブロックを、より広い探索範囲から選択し、その選択された前記複数の探索開始候補ブロックの中から最も相似度の高いものを探索開始ブロックとして特定し、その特定された探索開始ブロックを始点としてone-at-a-time探索を行い、これによって、前記着目するブロックと最も高い相似度を示す前フレームにおけるブロックを見つけることを特徴とする動きベクトル検出方法。

【請求項3】 着目するブロックと同じ位置にある前フレームのブロックについて相似度を求め、その相似度のレベルを所定の基準により判定する場合に、そのレベルに応じて3以上のクラス分けを行うことを特徴とする請求項2記載の動きベクトル検出方法。

【請求項4】 相似度の判定は、相似度が高いほど小さな値をとる関数を用いて行われ、前記関数は、絶対値差分または差分の2乗を算出する関数であることを特徴とする請求項2または請求項3のいずれかに記載の動きベクトル検出方法。

【請求項5】 請求項1～請求項4のいずれかに記載の方法により検出された動きベクトルを用いて動き補償を行い、動き補償された画像を用いてフレーム予測符号化を行う画像符号化方法。

【請求項6】 動き補償付きフレーム間予測符号化装置において、注目する現フレームのブロックと、このブロックと同位置にある前フレームのブロックとの間の相似度を算出する相似度算出手段と、算出された前記相似度の程度に応じて、動きベクトルの探索始点となるブロックの候補を選択し、その候補の中から探索始点となるブロックを選定する探索始点選定手段と、選定された探索始点ブロックを始点として動きベクトルの探索を行う動きベクトル探索手段と、を具備することを特徴とする動き補償付きフレーム間予測符号化装置。

【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、ブロックマッチング法を用いた動きベクトル検出方法および画像符号化方法に関する。

## 【0002】

【従来の技術】静止画像の場合、予測符号化に用いることができる参照画素はその画像内で、すでに符号化された画素に限られているが、動画の場合、1フレーム前またはそれ以前のフレーム中の画素を参照画素として用いることができ、予測効率の向上が期待できる。但し、前フレームの画素を用いた予測（フレーム間予測）符号化を行う場合、単純にフレーム間の差分を取ると前後のフレームで対応する点が大きくずれて予測の精度が低下する。

【0003】そこで、前後のフレームを比較して画像の動き（動きベクトル）を検出し、前フレームを、検出された動きに対応させてシフトさせて（つまり、フレーム間で生じた動きを補正して）フレーム間予測を行えば、差分のデータ量が減少して効率的な符号化を行うことができる。

【0004】図4（a）は、このような動き補償付き予測符号化を実行する符号化装置の概要を示している。動き補償部50は、前後のフレームの相似度の比較から動きベクトルを検出し、符号化対象のフレームの一つ前のフレームを、動きベクトルに対応させてシフトさせて動き補償を行う。フレーム予測符号化部51は、動き補償された画像と現フレームとの差分をとり、その差分データについて、例えば、離散コサイン変換（DCT）を用いた符号化を行う。

【0005】図4（b）は、フレーム52およびフレーム53の間で、動き補償を行わずに差をとった場合に、符号化するべき情報55がかなり大きなものとなることを示している。

【0006】図4（c）は、フレーム52とフレーム53との相似度の比較から動きベクトルVを求め、この動きベクトルVを用いてフレーム52をシフトさせ、この後にフレーム53との差をとった場合に、符号化するべき情報56が少ないことを示している。

【0007】動きベクトルを検出する一般的な方法としてはブロックマッチング法がある。ブロックマッチング法は、符号化対象フレームを複数のブロックに分割し、着目するブロックについて、一つ前のフレームの探索範囲内にあるブロックとの間の相似度を判定し、最も高い相似度を示すブロックの位置が、着目するブロックから  $(x, y)$  だけシフトしている場合に、そのシフト量  $v = (x, y)$  を動きベクトルとする方法である。

【0008】簡易な動きベクトルの検出方法としては、one-at-a-time探索がある。この探索方法は、注目するブロックに関し、1フレーム前の同じ位置にあるブロックを始点とし、水平および垂直方向にブロックを一つづ

つシフトさせながら相似度を探索していき、所定の探索範囲内で最も相似度の高いブロックを見つけだして動きベクトルを検出する方法である。この方法では、探索範囲と探索候補ブロックを限定するため、動きベクトル探索装置の構成を簡素化することができるというメリットがある。

#### 【0009】

【発明が解決しようとする課題】しかし、探索範囲と探索候補点を限定するone-at-a-time探索法においては、物体が高速に動いてフレーム間の差分が大きい場合（つまり、真の動きベクトルが大きい場合）、真のベクトルを検出する前に、狭い探索範囲内において局所的に条件が合致するブロックを、求めるブロックと判断してしまい、検出動作を終了してしまう場合がある。このように、一つのフレームについて動きベクトルの検出結果が適切でない場合は、そのフレームの画像復元の際、再生画像の画質劣化を招き、さらに、この画質劣化した画素を参照画素として次の復号化が行われるので、画質劣化が以降のフレームにも波及していく、という不都合が生じる。

【0010】本発明はこのような問題点を解決するためになされたものであり、真の動きベクトルが大きい場合でも、適切な動きベクトルを簡易に検出することができる動きベクトル検出方法を提供することを目的とする。

#### 【0011】

【課題を解決するための手段】本発明の動きベクトル検出方法は、注目するブロックと同じ位置にある前フレームのブロックとの間で相似度のレベルを判定し、その判定レベルに応じて、動きベクトルの探索を行う場合の探索範囲と探索候補のブロックをダイナミックに変更する。例えば、相似度が低い場合、注目するブロックは前フレームの遠い位置から来た可能性が高いので、探索の範囲を広げるとともに、探索候補のブロックをより遠くにとる。これにより、簡易な方法を用いて真の動きベクトルを効率的に検出することができる。

#### 【0012】

【発明の実施の形態】本発明の第1の態様にかかる動きベクトル検出方法では、現フレームの注目するブロックと、このブロックと同位置にある前フレームのブロックとの相似度を算出し、算出された相似度の程度に応じて、動きベクトルの探索開始候補となるブロックを変更する。

【0013】真の動きベクトルの大きさを推定し、そのベクトルの検出に適していると考えられる範囲内で適切なブロックを探索開始点として選定することにより、動きベクトルの大きさに追従可能な、適応化された動きベクトル検出方法が実現する。

【0014】本発明の第2の態様では、符号化対象フレームを複数のブロックに分割し、着目するブロックについて、前フレームの探索範囲内にあるブロックとの間の

相似度を判定し、最も高い相似度を示すブロックの位置が、前記着目するブロックから $(x, y)$ だけシフトしている場合に、そのシフト量 $v = (x, y)$ を動きベクトルとする、ブロックマッチング法を用いた動きベクトル検出方法において、前記着目するブロックと同じ位置にある前フレームのブロックについて相似度を求め、その相似度のレベルを所定の基準により判定し、相似度が低い場合には、相似度が高い場合に比べて、one-at-a-time探索の始点となる可能性をもつ複数の探索開始候補ブロックを、より広い探索範囲から選択し、その選択された前記複数の探索開始候補ブロックの中から最も相似度の高いものを探索開始ブロックとして特定し、その特定された探索開始ブロックを始点としてone-at-a-time探索を行い、これによって、前記着目するブロックと最も高い相似度を示す前フレームにおけるブロックを見つけ出すようにした。

【0015】同じ位置にある前ブロックとの間で相似度を検出し、その相似度のレベルを評価値として利用して動きベクトルの大きさを推定し、大きいと推定される場合にはone-at-a-time探索の範囲を広げて、より遠くのブロックを探索の候補に加えることによって、真の動きベクトルを検出する可能性が高まる。

【0016】本発明の第3の態様では、第2の態様において、着目するブロックと同じ位置にある前フレームのブロックについて相似度を求め、その相似度のレベルを所定の基準により判定する場合に、そのレベルに応じて3以上のクラス分けを行うようにした。

【0017】クラス分けに応じて、探索範囲と探索候補ブロックをダイナミックに変更することにより、画像の動きのスピードに対応した適切な探索を実行することができる。

【0018】本発明の第4の態様では、第2の態様または第3の態様において、相似度の判定は、相似度が高いほど小さな値をとる関数を用いて行われ、前記関数として、絶対値差分または差分の2乗を算出する関数を用いる。

【0019】これにより、簡易に相似度を判定できる。

【0020】本発明の第5の態様にかかる画像符号化方法は、第1の態様～第4の態様のいずれか記載の方法により検出された動きベクトルを用いて動き補償を行い、動き補償された画像を用いてフレーム予測符号化を行う画像符号化方法である。

【0021】これにより、動きの大きな画像と小さな画像が混在する動画であっても、適切な動きベクトルを効率的に検出して符号化処理を行う方法が提供される。

【0022】本発明の第6の態様にかかる動き補償付きフレーム間予測符号化装置は、注目する現フレームのブロックと、このブロックと同位置にある前フレームのブロックとの間の相似度を算出する相似度算出手段と、算出された前記相似度の程度に応じて、動きベクトルの探

索始点となるブロックの候補を選択し、その候補の中から探索始点となるブロックを選定する探索始点選定手段と、選定された探索始点ブロックを始点として動きベクトルの探索を行う動きベクトル探索手段と、を具備する。

【0023】これにより、動画像のスピードに追従して、適切に動きベクトルを検出して効率的な符号化を行える補償付きフレーム間予測符号化装置が実現される。

【0024】次に、本実施の形態におけるブロックマッチングの標記法および、動きベクトルの検出に用いるone-at-a-time探索の概要について、図5および図6を用いて説明する。

【0025】図5(a)の上側に示されるのは1つ前のフレーム100であり、下側に示されるのは現フレーム200である。ここで、図5(a)の下側に太線で示されるブロック(BLK1)に着目する。このブロック(BLK1)は4つの画素(b2, c2, f2, d2)を含む。

【0026】なお、この4つの画素のうちで、左上にある画素(b2)の座標を(i, j)とし、この画素(b2)の座標がブロック(BLK1)の座標を代表するものとする。また、「b2, c2, f2, d2」は4つの画素の参照符号であると共に、各画素の輝度値も表すものとする。以上の標記の規則は、他のブロックや画素についても同様に適用される。

【0027】ここで、着目するブロック(BLK1)が、前フレーム100のどの位置から来たのかを判定する場合を考える。

【0028】仮に、同じ位置(i, j)にあるブロック(BLK2)が、ブロック(BLK1)に対応するブロックであるとする、太い矢印で示されるベクトル120が求める動きベクトルとなる。この場合、シフト量がゼロなので、動きベクトル $v = (0, 0)$ となる。すなわち、前フレームと現フレームとの間で動きがない場合は動きベクトルはゼロベクトルとなる。

【0029】同様に、右に1画素分シフトした位置(i+1, j)にあるブロック(BLK3)に対応するブロックならば、ベクトル110が動きベクトルとなり、左に1画素分シフトした位置(i-1, j)にあるブロック(BLK4)に対応するブロックである場合には、ベクトル130が動きベクトルとなる。

【0030】着目するブロックに対応した前フレームにおけるブロックは、所定の探索範囲内でブロック間の相似度(D)を次式で計算し、最も高い相似度を示すブロックを調べることによって特定される。

【0031】

$D = \sum f(a)$  ; 但し、aは対応する画素間の差分値  
ここで、fは相似度を評価する関数であり、相似度が高いほど小さな値をとる関数である。このような関数としては、例えば、差分の絶対値を計算する関数や、差分の

2乗を計算する関数がある。本実施の形態では、絶対値差分をとる関数を用いる。

【0032】図5(b)は、ブロック間の相似度の判定(ブロックマッチング)に関して、本明細書における平面的な標記法の例が示される。

【0033】図5(b)において、「M1」は第1回目のマッチング判定を意味する。すなわち、着目するブロック(図5(a)のBLK1)と同じ位置(i, j)にある前フレームのブロック(図5(a)のBLK2)との間で、第1回目のマッチング判定を行うという意味である。

【0034】同様に、「M2」は第2回目のブロックマッチング判定を表す。この第2回目の判定は、(i-1, j)および(i+1, j)の位置にある前フレームの2つのブロック(図5(a)のBLK4, BLK3)との間で併行して実行される。

【0035】ブロック間のマッチング(相似度)は、上述のとおり、各ブロックの対応する画素間の輝度値の絶対値差分を加算することにより求められる。図5(b)の下側には、(i-1, j)の位置にあるブロック(BLK4)との間で相似度を求める場合の計算式が示されている。

【0036】図5(b)において、参照番号110および130は、図5(a)に示した動きベクトルである。動きベクトル120はゼロベクトルであるため、図5(b)の平面的な標記の場合には、記載されない。

【0037】以上がブロックマッチングの平面的な標記法についての説明である。

【0038】次に、図6を用いてone-at-a-time探索の概要を説明する。one-at-a-time探索は、探索範囲を限定して、1ブロック毎に探索を進めていく方法である。

【0039】図6に示すように、one-at-a-time探索では、まず、x方向(水平方向)に1ブロック毎に探索を進め、続いて、Y方向(垂直方向)に1ブロック毎に探索を進めていく。

【0040】最初に、(i, j)の位置にある前フレームのブロックとの間の相似度(絶対差分の和)を検出する(マッチング判定M1)。続いて、左右の隣接ブロック((i-1, j)および(i+1, j)の位置にあるブロック)との間の相似度を検出する(マッチング判定M2)。ここで、マッチング判定M1の結果が最小である場合には、前フレームと現フレームとの間で動きがないものと判断して、動きベクトルはゼロベクトルとなる。

【0041】一方、左右の隣接ブロックのどちらかにおいて絶対差分の和が最小となった場合、その最小値が得られたブロックの側に動きベクトルの始点となるブロックがあるものと推定する。例えば、右側のブロック(すなわち、(i+1, j)の位置にあるブロック)が最小値を示した場合、右側の方向に動きベクトルの始点とな

るブロックが存在すると推定する。そして、続いて、隣接する右側のブロックについて同様に絶対差分の和を計算する。

【0042】このような水平方向（X方向）のベクトル探索は、以下の① または② の条件が満たされるときに終了する。

① 算出した絶対差分の和が、前回の絶対値差分の和よりも大きくなる。

② 所定の探索範囲の外となる。

【0043】図6では、探索範囲内で、マッチング判定M3、マッチング判定M4と相似度の判定を進めてきた結果、マッチング判定M4の結果がM3の結果よりも大きくなり、したがって、このマッチング判定M3の対象となった $(i+2, j)$ の位置のブロックが水平方向のベクトル探索の結果として出力される。

【0044】続いて、 $(i+2, j)$ の位置にあるブロックを始点として、垂直方向のベクトル探索を行う。垂直方向のベクトル探索の方法は、水平方向のベクトル探索方法と同じである。

【0045】すなわち、始点となるブロックに隣接する上下のブロック（ $(i+2, j+1)$ 、 $(i+2, j-1)$ の各位置にあるブロック）について、5回目のマッチング判定M5を行う。ここで、マッチング判定M3の結果が最小値である場合は、このM3に対応した $(i+2, j)$ の位置のブロックが、求める動きベクトルの始点となるブロックとして検出される。

【0046】一方、隣接する上下のブロックのいずれかが最小値を示せば、そのブロックの方向に動きベクトルの始点となるブロックがあると推定する。図6では、 $(i+2, j+1)$ の位置にあるブロックが最小値を示したため、上側に向かってベクトル探索を進める。

【0047】すなわち、探索範囲内において、マッチング判定M6、M7を行った結果、マッチング判定M7の結果（絶対差分の和）がマッチング判定M6の結果よりも大きくなったので、M6に対応した、 $(i+2, j+2)$ の位置にあるブロックが、求める動きベクトルの始点を示すブロックとして検出される。したがって、図6中で太線で示されるような、動きベクトル「VA」が検出されたことになる。

【0048】以上がone-at-a-time探索の概要である。

【0049】このone-at-a-time探索は、例えば、探索範囲の中心（ゼロベクトル位置）から探索を開始する場合において、真の動きベクトルの始点がゼロベクトル位置の近傍にあるときには有効である。しかし、画像の動きが速く、真の動きベクトルの始点がゼロベクトル位置から遠い位置にある場合（すなわち、探索範囲の隅にある場合）には、その事実を検出する前に、上述の探索終了条件を形式的に満足するようなブロック（すなわち、局所的な最小値（「極小値」）を示すブロック）を検出して探索を終了してしまうことがある。

【0050】そこで、本実施の形態では、one-at-a-time探索の探索範囲と探索候補のブロックをダイナミックに変更し、真の動きベクトルを検出する可能性を高める。

【0051】以下、本発明の実施の形態について図面を参照して具体的に説明する。

【0052】（実施の形態）本実施の形態では、注目するブロックと同じ位置にある前フレームのブロックとの間で、各画素値の絶対差分の和をとり、これを評価値とする。

【0053】そして、その評価値のレベルに応じて複数のクラス分けを行い、そのクラス分けに応じて探索開始の候補となるブロック（ブロック群）を選択するようにする。探索開始の候補となるブロック群のパターンは、動きベクトルが大きい場合も考慮に入れて、予め定められている。

【0054】図1において、 $(i, j)$ の位置（座標の原点）にある前フレームのブロック（0）が、本来的に探索の始点となるブロックである。従来は、one-at-a-time探索の始点は常に、このブロック（0）であった。

【0055】しかし、本実施の形態では、図1（b）に示すように、 $(i, j)$ の位置にあるブロック間における相似度（各画素の絶対差分の和）を評価値として用いて、そのレベルに応じて、探索開始候補点（探索始点）を選択する。

【0056】つまり、差分値（評価値）が10以下の場合には、ブロック間の相似度が高いので、動きベクトルはゼロベクトルに近いと判断される。よって、従来どおり、原点（ $(i, j)$ の位置）を始点としてone-at-a-time探索を行う。

【0057】一方、差分値（評価値）が10より大きく100以下の場合には、動きベクトルは、それほど大きくないと判断される。つまり、現フレームの $(i, j)$ の位置にあるブロックは、前フレームのやや遠い位置から来た可能性が高い。したがって、真の動きベクトルを検出するためには、探索範囲を広くとる必要があり、しかも、探索を開始するブロック（探索始点）を原点から、やや離れた位置にとった方が、ブロック間の差分の最小値を与えるブロックを、的確かつ効率的に発見できる可能性が高い。

【0058】したがって、この場合には、one-at-a-time探索の始点となるブロックを、図1（a）のB群の中から選択する。図示されるように、B群は、原点を中心とした、81個（ $=9 \times 9$ ）のブロックを含む仮想的な正方形の4辺上において、4個おきに配置されたパターンをもつブロック群である。

【0059】そして、このB群のブロックの各々について、着目するブロックとの間で、絶対差分の差をとって相似度を算出し、差分値が最も小さいブロック（相似度が最も高いブロック）を、one-at-a-time探索の始点と

する。

【0060】また、差分値（評価値）が100より大きく1000以下の場合、動きベクトルは大きいと判断される。つまり、現フレームの（ $i$ ， $j$ ）の位置にあるブロックは、前フレームの遠い位置から来た可能性が高い。したがって、真の動きベクトルを検出するためには、探索範囲をさらに広くとる必要があり、しかも、探索を開始するブロック（探索始点）を原点から、さらに離れた位置にとった方が、ブロック間の差分の最小値を与えるブロックを的確かつ効率的に発見できる可能性が高い。

【0061】したがって、この場合には、one-at-a-time探索の始点となるブロックを、図1（a）のA群の中から選択する。図示されるように、A群は、原点を中心とした、289個（ $=17 \times 17$ ）のブロックを含む仮想的な正方形の4辺上において、8個おきに配置されたパターンをもつブロック群である。

【0062】そして、このA群のブロックの各々について、着目するブロックとの間で、絶対差分の差をとって相似度を算出し、差分値が最も小さいブロック（相似度が最も高いブロック）を、one-at-a-time探索の始点とする。

【0063】このようにして、本実施の形態では、評価値のレベルに応じて、one-at-a-time探索の範囲と探索始点がダイナミックに変更される。

【0064】図1（a）に示されるように、本実施の形態では、B群のブロックの中から、（ $i$ ， $j+4$ ）の位置にあるブロックが、探索始点となるブロックとして選択されている。そして、このブロックを始点として、one-at-a-time探索を開始し、水平方向のベクトル探索では（ $i+2$ ， $j+4$ ）の位置にあるブロックが検出され、続いて、垂直方向のベクトル探索において、（ $i+2$ ， $j+6$ ）の位置にあるブロックが検出される。この結果、図中、太線で示されるような、大きな動きベクトル「VA1」が、正確に検出される。

【0065】このような動きベクトルの検出を実行し、動き補償を行って符号化を行う装置の構成例が、図2に示されている。

【0066】動きベクトル探索部10は、ゼロベクトル評価値算出部11と、始点判定部12と、水平方向動きベクトル探索部13と、垂直方向動きベクトル探索部14と、を具備する。

【0067】ゼロベクトル評価値算出部11は、現フレームの対象ブロックと、これと同位置にある前フレームとの間で、各画素の輝度値の絶対差分の和をとってゼロベクトルの大きさ（評価値）を算出する。

【0068】始点判定部12は、評価値の大きさの程度に従って、上述のようなクラス分けを行い、探索始点となるブロック群を選択し、相似度の最も高いものを探索始点として選択する。

【0069】水平方向動きベクトル探索部13は、水平方向にone-at-a-time探索を行って、水平方向の動きベクトルを検出する。

【0070】垂直方向動きベクトル探索部14は、垂直方向にone-at-a-time探索を行って、垂直方向の動きベクトルを検出する。この結果、動きベクトルが求められる。

【0071】動き補償部21は、検出された動きベクトルを用いて動き補償を行った後、前フレームの各画素との差分を算出する。算出された差分値が、符号化すべき、現フレームのデータとなる。

【0072】離散コサイン変換部30は、差分値で表された現フレームについて離散コサイン変換（DCT）を行い、符号化する。

【0073】図2の符号化装置の動作手順を、図3に示す。

【0074】まず、ゼロベクトル評価値算出部11がゼロベクトルの大きさ（すなわち、同じ位置にある前フレームのブロックとの差分値）を算出し、これを評価値とする（ステップ40）。

【0075】次に、評価値の程度から、探索開始候補点（探索始点）の変更が必要かどうかを判定する（ステップ41）。そして、評価値が10以下のときは、原則どおり、原点を探索始点とする（ステップ42）。

【0076】一方、評価値が100以下の場合、探索始点の候補点として、第1群のブロック（図1ではB群のブロック）を選択し（ステップ43）、1000以下の場合には第2群のブロック（図1ではA群のブロック）を選択する（ステップ44）。次に、探索開始候補のブロック群の中から最小差分を与えるブロックを探索始点とする（ステップ45）。

【0077】続いて、求められたブロックを始点としてone-at-a-time探索を行い、水平方向のベクトルを検出する（ステップ46）。続いて、垂直方向のベクトル探索を行って（ステップ47）、動きベクトルを検出する（ステップ48）。

【0078】以上説明したように、本実施の形態では、ゼロベクトル点での絶対差分和の大きさ（評価値）に基づき、真の動きベクトルの大きさを予測し、one-at-a-time探索を開始する始点を決定するように構成した。

【0079】そして、真の動きベクトルがほとんどゼロのときには、ゼロベクトル点からone-at-a-time探索を開始し、真の動きベクトルが小さいときには、第1群のブロック群を探索し、真の動きベクトルが大きいときには、第2群のブロック群を探索することにより、真の動きベクトルが変化するような動画像においても、効果的に動きベクトルを検出することができる。よって、画質劣化をもたらさずに効率的な符号化を行える。

【0080】なお、上述の動きベクトル検出方法では、one-at-a-time探索を開始する候補点を、原点、第1群

および第2のブロック群の3つの候補の中から選択する構成であるが、これに限定されるものではなく、4つ以上の候補から選択することもできる。評価値に基づくクラス分けを多く行えば、真の動きベクトルを効率的に検出できる可能性が高まる。

【0081】また、探索始点候補のブロック（ブロック群）のパターンも、上述の例にとらわれることなく、自由に選択できる。例えば、第1群を構成するブロックの数は8個でなくてもよいし、他の配置で構成されてもよい。

【0082】さらに、one-at-a-time探索を探索手法に用いているが、他の簡易な探索手法による場合でも、本発明は適宜、変形して適用が可能である。また、上述の例では、絶対差分和を評価値としているが、二乗誤差和等の他の評価値を採用することも可能である。

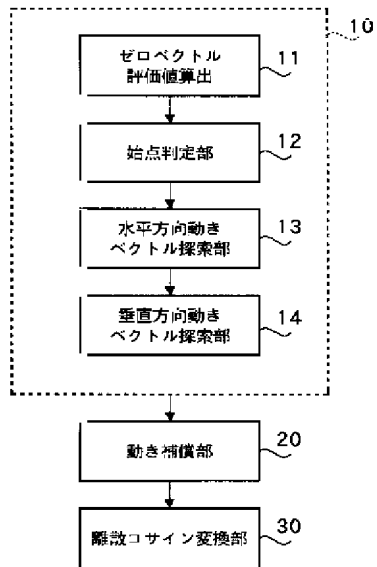
【0083】

【発明の効果】以上説明したように本発明によれば、評価値によって選択的に動きベクトル探索範囲と探索候補点を選択するので、真の動きベクトルが大きい画像と小さい画像が混在する動画画像にあっても、適切な動きベクトルを簡単な方法で得ることができるという有利な効果が得られる。これによって、高精度のフレーム間予測符号化を行うことができる。

【図面の簡単な説明】

【図1】（a）本発明の一実施の形態にかかる動きベクトル検出方法を説明するための図

【図2】



（b）評価値による探索開始候補のブロック群選定の例を示す図

【図2】本発明の一実施の形態にかかる動きベクトル検出装置の構成例を示す図

【図3】動きベクトル検出装置の動作手順を示すフロー図

【図4】（a）動き補償付きフレーム予測符号化の手順を説明するための図

（b）動き予測を行わない場合の符号化すべき情報量を示す図

（c）動き予測を行う場合の符号化すべき情報量を示す図

【図5】（a）ブロックマッチングを用いた動きベクトルの検出方法を説明するための図

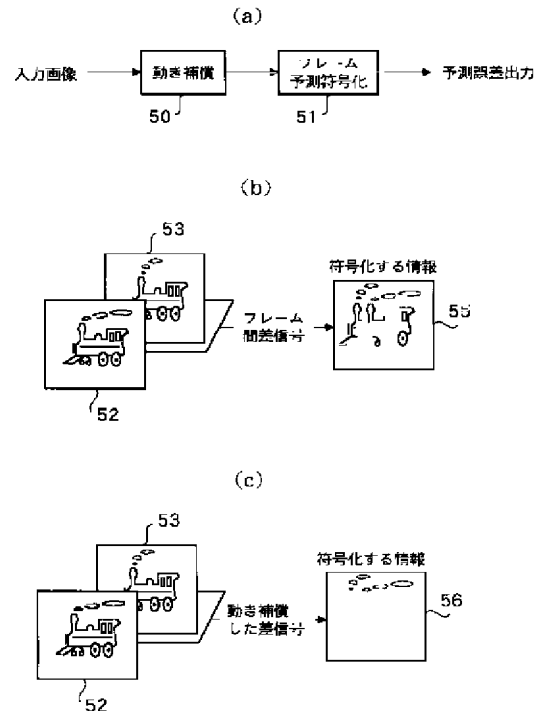
（b）動きベクトルの検出方法の、平面的な標記法を説明するための図

【図6】one-at-a-time探索法の概要を説明するための図

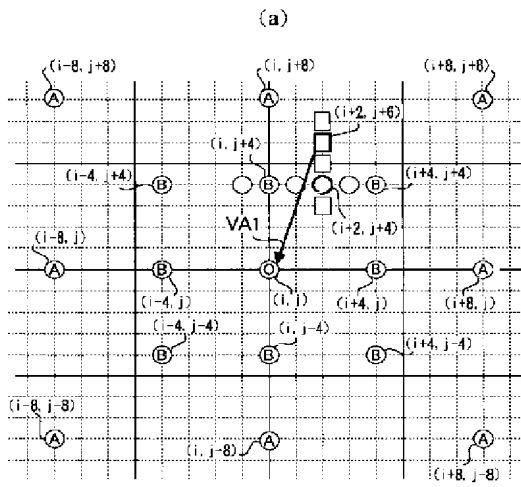
【符号の説明】

- 10 動きベクトル検出装置
- 11 ゼロベクトル算出部
- 12 始点判定部
- 13 水平方向動きベクトル探索部
- 14 垂直方向動きベクトル探索部
- 20 動き補償部
- 30 離散コサイン変換部

【図4】



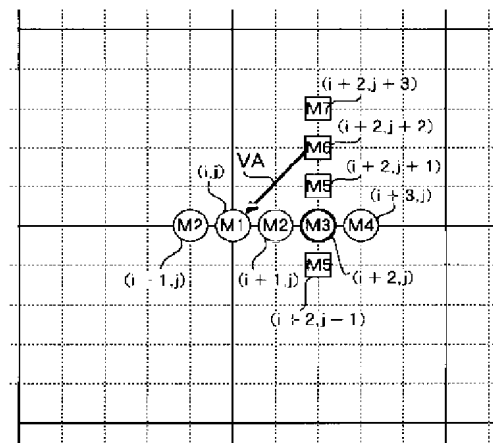
【図1】



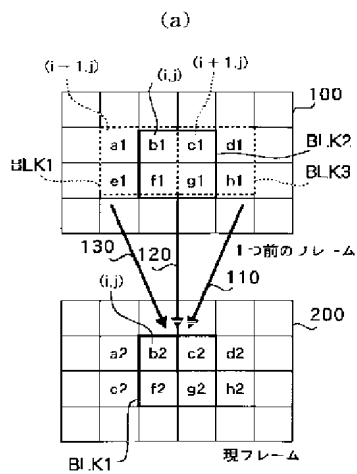
(b)

| 差分値    | 探索開始候補 |
|--------|--------|
| 10以下   | 原点     |
| 100以下  | B群     |
| 1000以下 | A群     |

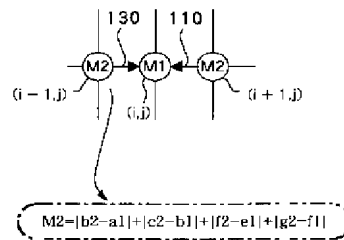
【図6】



【図5】

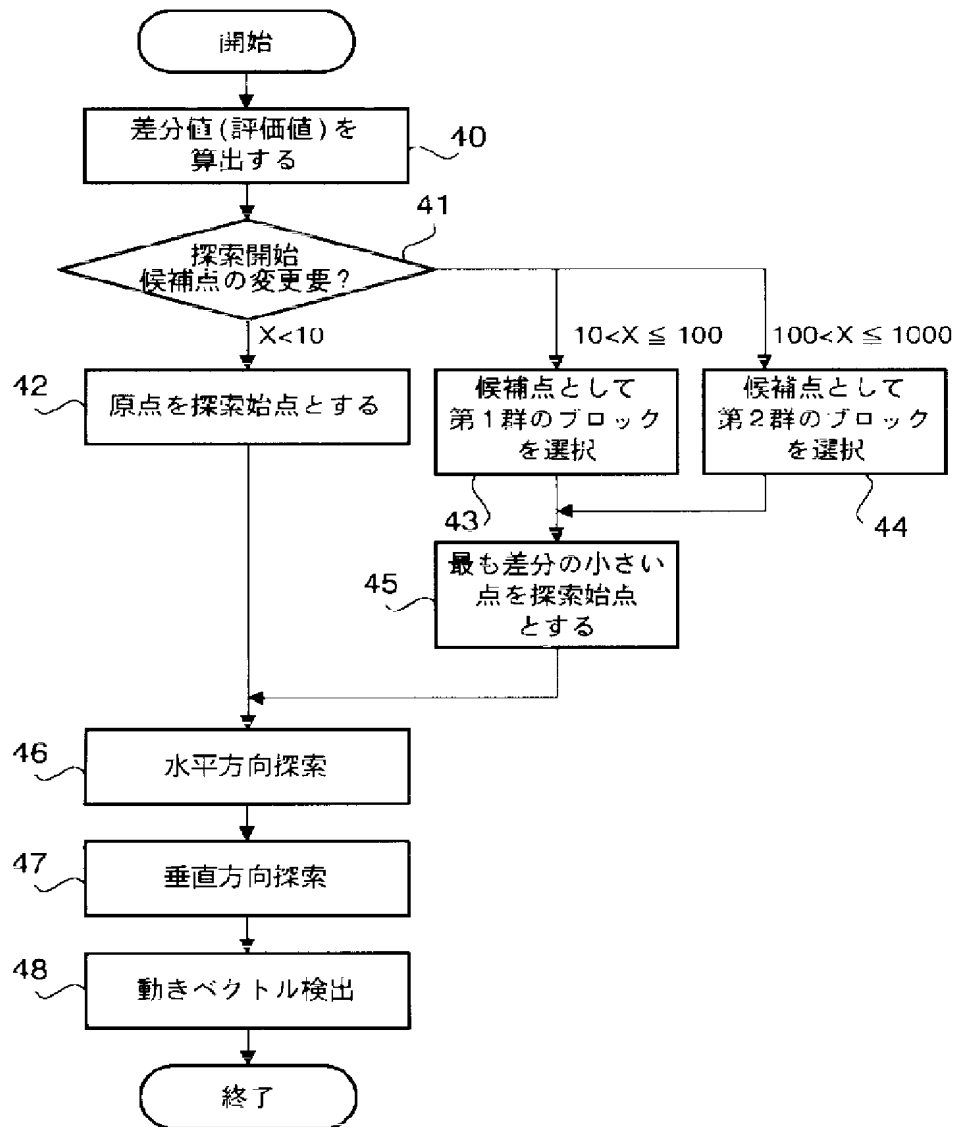


(b)





【図3】



# MOVING VECTOR DETECTING METHOD AND IMAGE ENCODING METHOD

**Publication number:** JP2000152244 (A)

**Publication date:** 2000-05-30

**Inventor(s):** IDO TAIJI +

**Applicant(s):** MATSUSHITA ELECTRIC IND CO LTD +

**Classification:**

- international: **H04N7/32**; (IPC1-7): H04N7/32

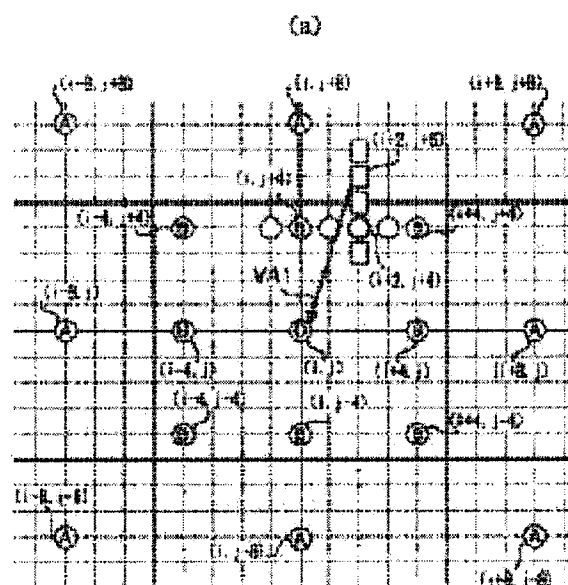
- European:

**Application number:** JP19980320760 19981111

**Priority number(s):** JP19980320760 19981111

## Abstract of JP 2000152244 (A)

**PROBLEM TO BE SOLVED:** To easily detect a proper moving vector even when a real moving vector is large by calculating the similarity between a block of interest in a current frame and the block in the last frame at the same position with the block of interest and changing a block as a search candidate for the moving vector. **SOLUTION:** The similarity between blocks at positions (i,j) is used as an evaluated value and a search start candidate point is selected according to its level. Namely, when the difference value (evaluated value) is  $\leq 10$ , a block as the start point of a one-at-a-time search is selected from a B group. Then similarity is calculated as to respective blocks of the B group by calculating the absolute difference from the block of interest and the block having the least difference value is regarded as the start point of the one-at-a-time search. When the difference is 100 to 1000, the block as the start point of a one-at-a-time search is selected from an A group.



(b)

| 差分値    | 検索開始候補 |
|--------|--------|
| 10以下   | 源点     |
| 100以下  | B群     |
| 1000以下 | A群     |

Data supplied from the **espacenet** database — Worldwide